

## MEGABUG

## INTERACTIVE MACHINE CODE DEBUGGING TOOL

R. Oppenheimer

## GENERAL

Megabug is a fully interactive machine code debugging system which is written for the Interak computer. It will present you with a front panel type display of the Z80's registers which can be modified using cursor commands. You can then execute a program either continuously or in single step mode under the control of Megabug. When running a program under Megabug you can interrupt it at any point simply by pressing the space bar and you will be presented with the Z80 register panel. This is an especially useful feature if you are debugging a program which expects input from the keyboard- just wait for the prompt to appear then interrupt using the space bar and alter the registers as necessary.

Megabug will work on programs which operate on the Interak screen as it uses an auxiliary screen memory area which is copied to the VDU when the object program (the program being debugged) is executing and back again when Megabug needs it. For this reason Megabug is transparent to any program running under it. It will also work on ROM based programs as it uses a dynamic program execution technique rather than the more familiar method of inserting breakpoints into the actual program code. It also does not require any hardware modifications as it does not need to produce interrupts to perform its single stepping function.

## TERMS USED

Certain terms have been used throughout this documentation and they are explained as follows.

Object program	The program being debugged
Program screen	The display present on the VDU when the object program is running. This will be the same as if Megabug was not present.
Command screen	The display present on the VDU when the Z80 registers are being displayed and Megabug is ready to take commands from the keyboard.

## SYSTEM REQUIREMENTS

Megabug requires a working Interak computer with a 64 char/line VDU occupying addresses F000 to F5FF and an ASCII keyboard on port 40H using bit 7 as a strobe.

It will support an optional printer using port 7 as the output data and port 6 as the status port (bit 7 is the ready flag)

Note: the "P" command will not work unless the printer is installed and on line.

Megabug resides in memory from B000 to B8E2 and uses memory up to BFFF as workspace and auxiliary screen memory.

#### RUNNING MEGABUG

The program is supplied on tape which will load under the Zymon monitor to addresses B000-B8E2.

Having loaded it execute Megabug from its coldstart address. When the command screen appears set the object program "PC" register to the address the object program is to run from and enter it using the <CR> command.

Whenever the command screen is displayed all the Megabug commands will be available to you.

If you need to exit from Megabug at any point to use the system monitor to inspect memory areas (for example) use the Q(uit) command. The program screen will be saved together with the status of the object program where you left off. It is possible to resume where you left off by executing a warm start.

Cold start address      B000  
Warm start address      B002

#### THE REGISTER DISPLAY

The display is divided into nine fields each under its own heading. They are as follows.

ADDR                    the address of the first byte of the instruction appearing on that line.

INSTRUCTION            a dis-assembly of the instruction executed in the current line.

BRPT                    an address which will cause execution of the object program to be interrupted if it is running under the "R" command when BRPT is encountered.

PC,AF,BC,DE            the contents of the Z80 registers after execution  
HL,IX,IY,SP            of the current instruction.

Note: the registers and BRPT can be altered using the Megabug commands, ADDR cannot.

## SYSTEM COMMANDS

### HEX characters 0-9,A-F

These are echoed directly to the cursor position and are used to alter the register display. You will notice that the cursor positioning is automatic in this mode so it is impossible to enter a character in an illegal position.

### <CR> (carriage return)

<CR> will enter an updated register line after it has been changed using the HEX character keys. When you do this the old values of the registers will be reprinted to remind you of what they were before you altered them and your revised version will be printed underneath. The lower line will have the instruction field left blank to indicate that they are user input values. Unless <CR> is used after altering the registers any changes made will be ignored at the next execution of an object program instruction and the display will revert to its original state.

### ctrl-S,ctrl-D

These keys move the cursor left or right respectively by one hex number. They have no other effect and are intended to enable you to position the cursor rapidly on the registers that you may wish to change. Ctrl-S will position the cursor on the beginning of the current number if it is not already at the start of a number (in which case it moves to the beginning of the previous number). If you make an error whilst entering a number you will have to use ctrl-S and re-enter the whole of that number again.

### ctrl-C

This will restore the register display to what it was prior to any alterations made since the last <CR>. This is useful if you become confused when making changes to the registers and want to start afresh.

### V

V will display the program screen to allow you to inspect the effect that the object program is having on the screen. Pressing any other key will cause the screen to revert to the command screen with the cursor unmoved.

### P

This printer command causes the screen output to be echoed to the printer. It will immediately cause a heading to be printed on the paper similar to that at the top of the command screen in readiness for output of information. When the printer option is set a "P" will show on the command screen heading to show that output will be echoed to the printer. To cancel the P option press "P" again and the flag will clear.

Note: When setting or using this option it is essential that the printer is on-line or Megabug will wait indefinitely until it is.

## Q

This causes a quit to the system monitor by way of a jump to 0000H. The contents of the program screen (as displayed by V) and the current instruction with its corresponding register values will be preserved in case of a warm start some time later which will cause Megabug to carry on where it left off.

## R

R makes the object program start running starting from the point as displayed on the command screen. Execution will be many times slower than normal as Megabug will be running and executing the object program instructions one at a time under its supervision. This means that execution of the object program instructions will only account for a tiny fraction of the total processor time.

Program execution will stop if:

- a) The program has just executed an instruction whose start and finish addresses include the address under BRPT on the command screen.
- b) The space bar is pressed.

When the object program is running the program screen is displayed so you can watch the effect it is having on the screen - and stop it if you see something you don't like!

When program execution is halted you are free to use any of the commands normally available to you.

## <SPACE>

Pressing the space bar will cause the object program to be single stepped from the point last displayed on the command screen. When you execute a single step you may notice the screen give a slight flicker. This is because it is momentarily displaying the program screen whilst the object program is running in case it needs it.

Note: Both the <SPACE> and the "R" command will clear any alterations you have made to the registers since the last <CR>. In other words "R" or <SPACE> implies "ctrl-C" preceeding it. The command screen is corrected for this to prevent any confusion.

## DEBUGGING HINTS

a) If the object program includes a loop and you want to look at the registers after each loop then set a breakpoint at a suitable point in the loop. Pressing "R" will cause a new execution of the loop each time it is pressed.

b) If you need to look at some area of memory during program execution then use "Q" to go to the system monitor and use that to inspect the memory. When you want to continue program execution do a warm start and you will be able to continue running the object program from where you left off. (You will

have to reset the "P" option if you were using it.)

c) If your program requires a user input then you can use the "R" command and when you see the prompt appear on the screen interrupt using the <SPACE>. You should now be in a keyboard polling loop so you can single step until you reach the appropriate instruction e.g. IN A,(40) and then alter the registers appropriately. Using the "R" command will cause resumption of program execution. This method will avoid disturbing any breakpoints that you may have set, however if the keyboard polling loop is a long one you may find it more convenient to set a breakpoint at the appropriate instruction to avoid having to single step excessively.

d) Using the system monitor to modify addresses within the object program is not recommended if you are using an assembler to produce the object code. If you do then the source and object codes will not correspond. However if you are writing directly in machine code then Megabug will be a valuable development tool.

e) It is quite possible to go back a number of steps using Megabug. Just change the program counter and other registers to what they were at that former stage. However, beware in case you execute an instruction which changes a memory value e.g. INC (HL) or LDI as executing it again may cause problems so use this facility intelligently.

## CONCLUSION

Megabug is a valuable addition to your machine code tool kit. It can be used either in the development of your own programs or to investigate the workings of any machine code program which you possess. In either case Megabug will make debugging machine code programs both simpler and quicker.